

**REPRESENTATION OF RYHTHMIC MOVEMENTS CONSTRUCTED
USING A 3D MODEL SIMULATION**

A thesis submitted for partial fulfillment
of requirements for graduation with distinction
for the degree of Bachelor of Science
in the Department of Electrical and Computer Engineering
of The Ohio State University

By

Charelle JaNeen Wright

The Ohio State University
March 2007

Honors Examination Committee:

Professor Hooshang Hemami, Advisor

Professor Bradley Clymer

Approved By:

Advisor

Electrical and Computer Engineering

Copyright

Charelle Wright

© 2007

ABSTRACT

The purpose of this project is to emulate with a human model representation of how the body reacts, with synchronous movements, to interpreted music. With this project I plan to gain a general understanding of the brains activity but with a greater focus on how the body actually reacts to the underlying beat of the music for periodic movements. I will visually represent the human body with a constructed 3D model, using Coin3D (a 3D modeling software). After successful completion of the modeling program I plan to generate periodic signals, consisting of sine and cosine functions, in Matlab and import the files into my modeling program, saving the values in the predetermined joint arrays (transform nodes). This will then cause the 3D model to move with similar movements to that of the human body after the brain has interpreted the music and sent out the necessary signals throughout the body.

Dedicated to my loving family

ACKNOWLEDGEMENTS

I would first like to thank my advisor, Hooshang Hemami, for providing me the great opportunity to conduct research under his mentoring. Thanks for keeping me on track and focused, especially when I would make the project harder than needed. I would also like to thank Professor Bradley Clymer for walking me through the honors distinction project details and for also being a reliable resource for my project. Next I would like to thank Dave Donley for keeping me on target with the necessary documents and procedures I needed to follow and also for his patience and his ability to keep faith in my abilities to complete this project.

I also would like to thank Mansoor Kahn, a graduate student here at The Ohio State University, for educating and coaching me through the programming portion of my project. Your volunteering efforts are greatly appreciated.

I would also like to thank my family for all of their moral support during this very demanding time of my life.

VITA

October 16, 1984Born – Dayton, OH

2002.....Graduated from Meadowdale High School

2002-present.....Undergraduate Student
Department of Electrical and Computer Engineering
The Ohio State University

March 18, 2007.....B.S. Electrical and Computer Engineering
The Ohio State University

Table of Contents

	Page
Abstract.....	iii
Dedication.....	iv
Acknowledgements.....	v
Vita.....	vi
List of Figures.....	ix
Chapters:	
1. Introduction.....	1
1.1 Motivation.....	1
1.2 Research topic?.....	4
1.3 Objectives.....	5
2. Research Review.....	6
2.1 Previous Research.....	6
3. 3D Model Simulation Design.....	8
3.1 Preliminary Steps.....	8
3.2 Software/system Design.....	11
3.3 Design of body.....	14
3.4 Generating joint angles.....	17
4. Experimental Results.....	20
4.1 Results.....	20
4.2 Complications.....	22

5.	Conclusion/Recommendations.....	24
	5.1 Summary of completed work.....	24
	5.2 Possible future work.....	25

Bibliography

Appendix

- A Coin3D program used to design 3D body
- B Matlab code used to generate joint angles
- C Sample of joint variable inputs into transform nodes

LIST OF FIGURES

Figure 1: First quarter Gantt chart.....	8
Figure 2: Second quarter Gantt chart.....	10
Figure 3: System outline.....	12
Figure 4: Structural system of body in program terms	14
Figure 5: Body Layout.....	16
Figure 6: Generated signals for joints 1, 2, & 3.....	18
Figure 7: Generated signals for joints 4, 5, & 6.....	19
Figure 8: Actual 3D body model.....	21

CHAPTER 1

INTRODUCTION

1.1 Motivation

Within the last few decades there has been an explosion of interest regarding the brain and the affected that music have on the brain. Numerous questions and mysteries have sprung up among many neuroscientists and nonscientists. And now that the technology is increasing these curious individuals are slowly becoming more knowledgeable about the type of affected music have on the brain. Even though at times these findings from various scientists seem to contradict, the underlying point is that music definitely has an affected on the brain and the body. As time goes on there are new forms and genres of music that is developing in the western culture and with these new forms of music an indescribable event of activities have begin to take place. Violent acts of crime and wrong doing is now being blamed on the type of music an individual was listening to. In most cases the individual, usually an adolescent, will say that they were only trying to emulate what they heard in the music lyrics. Can music really have that type of affected on the brain? Some argue yes while others seem to disagree. Another situation dealing with music and how it affects the brain in a more positive aspect is the ability of how listening to music while studying can increase the amount of information

you can retain and remember. Some researchers suggest that if you study with calming and soothing music, which relaxes the mind, you may be able to retain more of the information you are studying and if you study while listening to this soothing music, you will perform better during test taking time if you can take the test under the same conditions of which you studied for the test (in this situation that is listening to the same type of soothing music). Now these issues are rather in depth and are more suitable for a researcher whose background is in neurology or sociology. There are a numerous amount of other topics and questions individuals (nonscientists included) have regarding this issue with the affected music has on the brain but the biggest problem is that these individuals that aren't scientists have a hard time understanding the scientific journals that are published, so one of my objectives for this project is to develop a tool and provide information that the common person could use and understand in their search to better understand the affected of music on the brain.

In my case I am an undergraduate student studying Electrical and Computer Engineering, so I had to think of a pertinent topic that I could research and study that both met my needs and was suitable for me to complete based off my resources and current area of study. So my next steps were to brainstorm various topics that fit both my needs and met the criteria for what was a realistic project for me to research as an undergraduate student. More information about me that is pertinent in why I am so interested in this topic is that I love and appreciate all genres of music and for the past

four years I have been a professional Disc Jockey. I figured if I was going to conduct research that my research topic be something that was interesting to me. So I figured being able to combine my two passions together and creating a research topic with the combination would be great! And if I could think of a topic that intertwined my passions for both music and electrical engineering it would be a once in a lifetime experience and interesting challenge for me to undergo.

1.2 Research topic?

The process of determining my research topic was one of the most difficult steps I had to complete because I was limited on the time I had available, which was two quarters of school, and I was limited to the resources that were available to me. Since I am an electrical and computer engineer with a specialization in power and control, I decided to work with my controls professor, Hooshang Hemami, on my honors distinction project because I was interested in learning more about the type of research a controls engineer my conduct. After getting approved to work with Professor Hemami on my research project I began to brainstorm ideas on possible research projects that dealt with my interest on how music affected the brain. During these brainstorming sessions I brought up the topic regarding how the brain is capable of interpreting music and then send this signal throughout the body for synchronous movements. With this idea I could construct a model of the human body as a functioning system, then sending a music signal to the system and have the body react to the various sent signals. My professor agreed with the ability of me being able to complete a research topic of this nature but he continuously stressed the importance that I didn't try to indulge in explaining how the brain actually interpreted the music and then sent it to the body but rather focus on completing the simulation of the system (the human body) and how it responded to the input signals, which would act as input music signals.

1.3 Objectives

My objectives for this project were rather simple. The first was to provide a simulation model that would help in aiding researching on learning more about how the brain sent an output from the music it interpreted to the rest of the body, which resulted in synchronous movements. From experience in the engineering field and the increase in technology, I have learned that it is more cost effective and safe to start your preliminary testing on a simulation or model. Once you have worked out all criteria for testing, you can then move on to a larger scale test that may include human subjects in this case of researching how the brain interrupts music and sends this signal throughout the body. It would be devastating to harm your human subject because you ran a series of test that were detrimental to the human brain or body but running your tests first with a model will help eliminate the probability of the prior statement occurring.

The second objective is for me to provide information that is usable and understandable to the average person. There are some scientific journals available that discuss this topic but the published text is so technical that the average, nonscientist, is unable to follow the vernacular of the published works. With this research I want my audience to be the average person, that way more individuals can benefit from the research I conducted and the simulation I developed. What good would it be for me to spend so much time on research and development if only a small body of the population could relate or benefit from what I was able to accomplish.

CHAPTER 2

REASEARCH REVIEW

2.1 Previous Research

Since my research does not consist of me conducting actual research on how the actual human brain interprets music, I figured it would be beneficial for me to first illustrate some of the recent findings from scientists around the world. I am not an expert on this topic and cannot determine the accurateness of these findings. Therefore I prefer that you not completely depend on these statements alone but rather you use them as an aide while conducting your own experiments regarding these various findings and developments.

Babies don't know much regarding the science and techniques of music but at that early age they do have a musical preference. This statement is backed up by previous researchers who conducted research on a similar topic, neuromusical research. Under this topic past researcher have been able to pinpoint various points regarding music and the brain. Some of these points are:

1. The human brain has the ability to respond to and participate in music.
2. The musical brain operates at birth and persists throughout life.
3. Early and ongoing musical training affects the organization of the musical

brain.

4. The musical brain consists of extensive neural systems involving widely distributed, but locally specialized regions of the brain:

- Cognitive components
- Affective components
- Motor components

5. The musical brain is highly resilient.

This information was provided solely for you to establish a basis of where the world is in its development and learning stages of how the brain is affected by music and not as a sole purpose of educating you on these topics (please refer to the disclaimer made in the previous paragraph). As I stated these are only my findings from my research and I did not personally conduct research on these particular topics and issues.

CHAPTER 3

3D MODEL SIMULATION DESIGN

3.1 Preliminary Steps

Now that I was successfully able to select a research topic, my next steps were to design a timeline for the duration of this project. I used a simple Gantt chart for this step in my process. Below is a copy of the Gantt chart I used for the first quarter I worked on my research project.

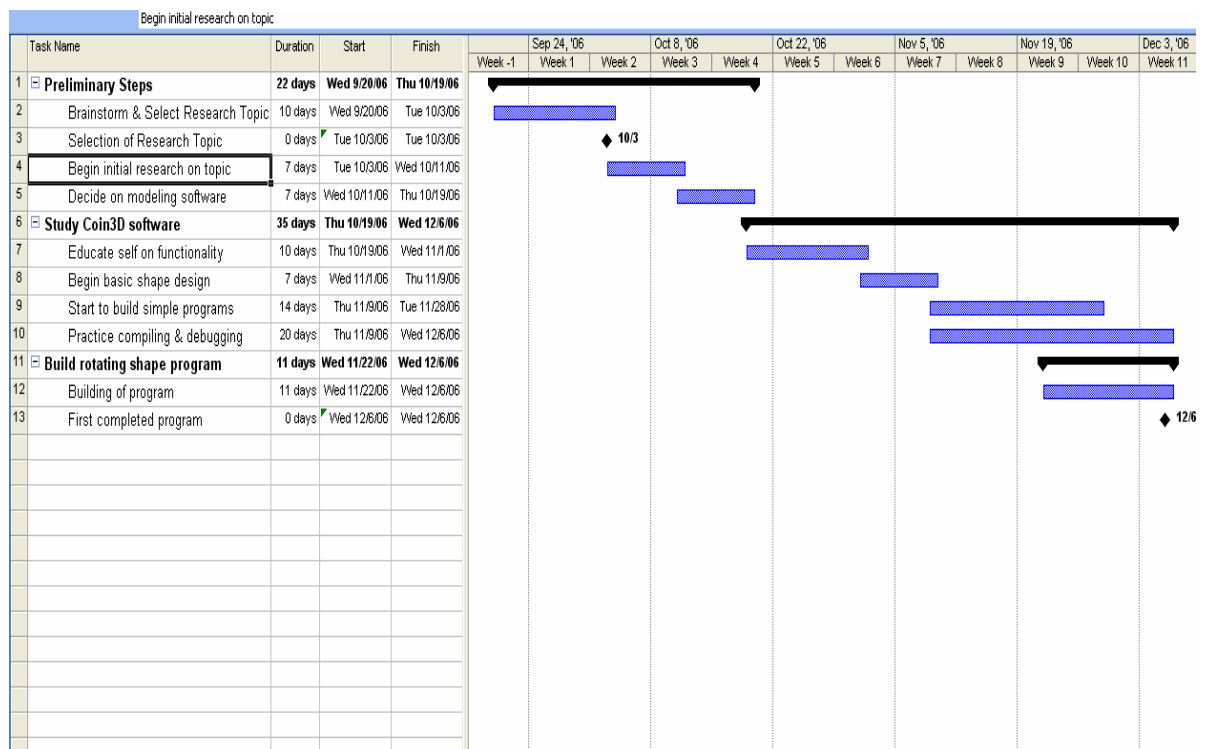


Figure 1: First quarter Gantt chart

I realized that time was an important constraint in my case, since I only had two quarters to complete my research project, and since I wanted to ensure my completion of my project I figured using a Gantt chart would assist me in staying focused and completing each step in a timely manner. If I was going to start this research I definitely had to prioritize my time effectively and this chart aided me in doing exactly that. Due to the small text of Figure 1, I will go into some detail of what this Gantt chart displays. This chart is broken down into three main sections, including two major milestones. The first section consists of the preliminary steps which are: brainstorming and selecting research topic (first major milestone), and beginning of initial research on topic as well as available 3D modeling software. Next is the section regarding studying the functionality of the Coin3D software that I selected for the modeling of my 3D human body, which consisted of learning functionality and commands of program, as well as constructing simple programs that specified various (i.e. cylinders, spheres, etc.) shapes. The last section consisted of me actually putting together a program that enabled a shape of my choice to rotate. This was an important stage because once I wanted the body to react to the input music signals the shapes will have to rotate about some axis and thus why it was my second milestone.

Below is the Gantt chart I used for the second quarter of my research project which consisted of three major sections, which are complete the 3D model program, complete the Honors thesis, and last to prepare and present my Oral Defense.

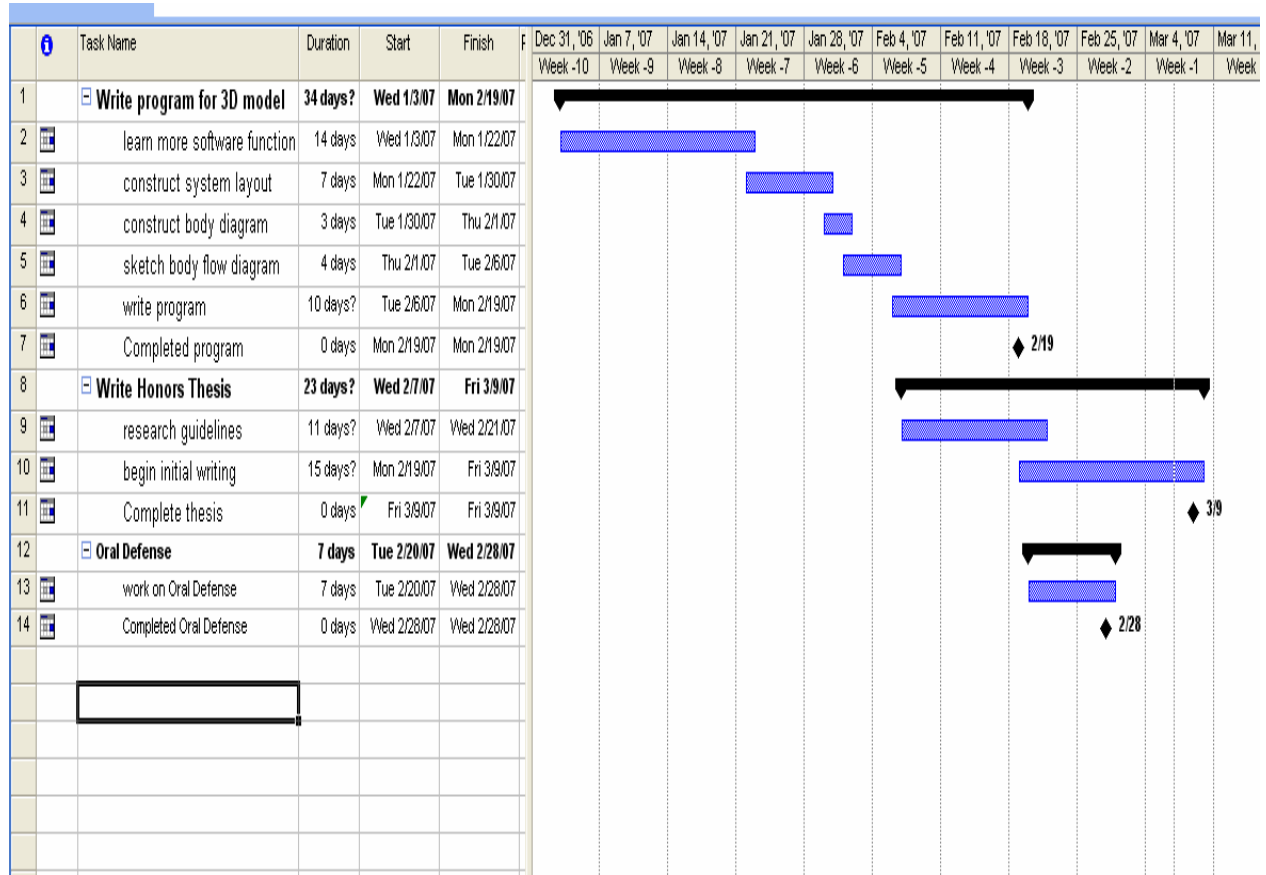


Figure 2: Second quarter Gantt chart

Now that I have outlined all the preliminary steps and described the two different Gantt charts I used throughout the duration of this project I will now give more detail regarding each step in the following sections.

3.2 Software/system Design

Prior to putting together the program that would build my 3D human body, I figured it would be beneficial for me to first outline the complete system. That way I could always have something to refer to and not have to depend solely on an mental picture. This system outline integrated every aspect of this project pertaining to the actual programming and was a great visual aide to graphically show what needed to be done and how all the components needed to be tied together in the end for accurate functionality of the program. Below is the system outline that I constructed and referred to throughout the remainder of this project.

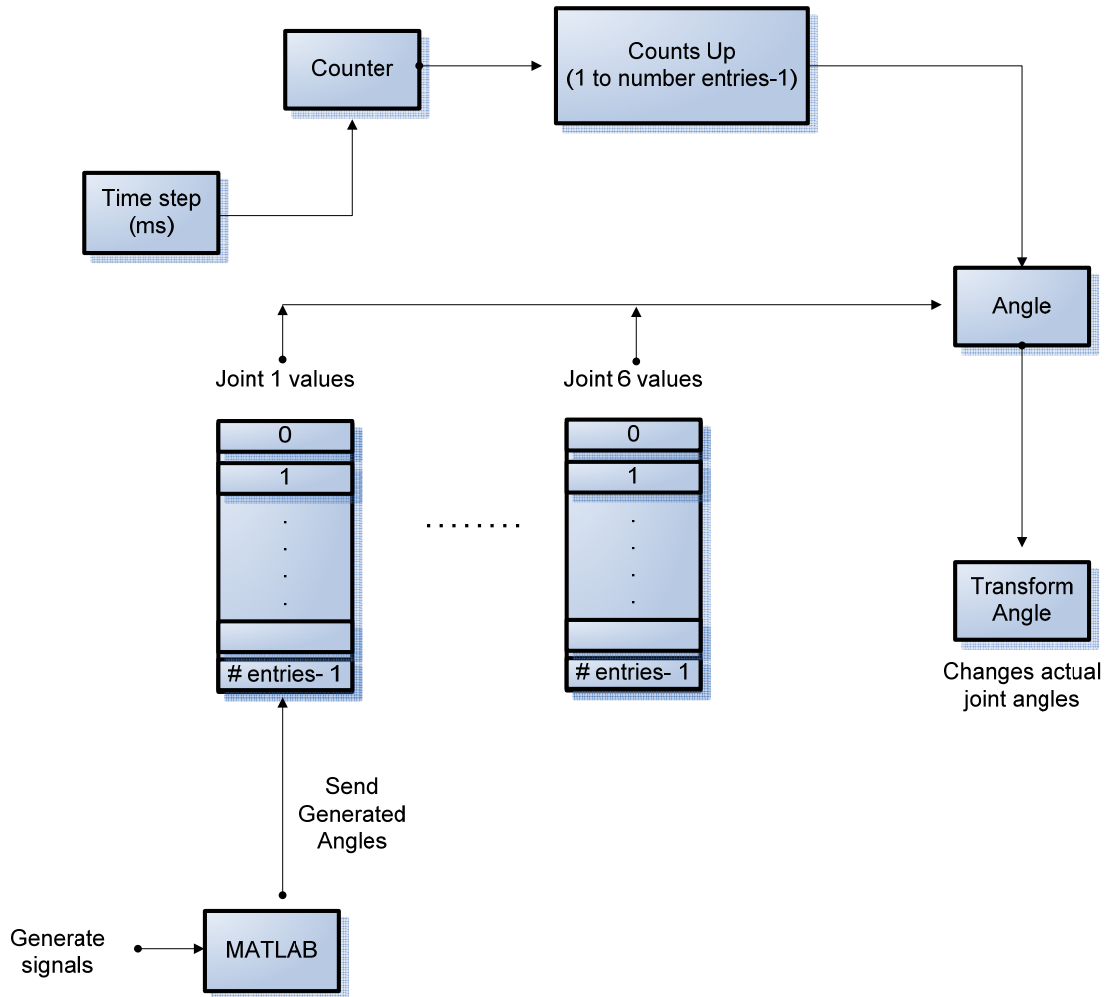


Figure 3: System outline

I am going start explaining this diagram with the counter in the top left hand corner. It was important to build a counter in the program because this function was responsible for moving the array position within the joint value arrays. A key input to this component was the actual time step, which controlled the speed of the counter. As the counter counted up at the speed specified in the time step the position vector inside the

joint arrays would cycle through the array pulling out the specified joint angle values within the array. Once the joint angle was selected from the joint arrays it would be sent to update the angle variable, which would then be sent to the dedicated transform node in the program that was responsible for moving, one of the six, necessary joints. Now focusing on the bottom left corner of the diagram you notice this is where you have to generate the actual joint angles that will be stored in the joint angle arrays. This portion of the system can be implemented multiple ways including: actually having a music string fed into Matlab and is then filtered to have an output of various signals or you could generate your own sinusoidal type signals and then send and store these values in the specified joint angles. For the purpose of this project I constructed six different sinusoidal signals, which were constructed of sine, cosine and tangent functions, and stored them in their designated joint angle arrays. These joint angles had to be generated and stored in the arrays before the counter could pull the various values from the arrays. Also determining the amount of angle points I wanted to enter was completely arbitrary, so I decided to have multiple runs changing the number of angle entries from 100 to 1000 and documented the change. When changing the amount of angle entries I noticed that I would also have to decrease the frequency of the joints in the program. Now that the actual system was outlined I could begin the actual programming of the body and the Matlab program that would generated the necessary joint angles.

3.3 Design of Body

Now that some of the preliminary work was completed I could begin the actual program that would visually represent my 3D human body. I first decided to make a sketch of how I wanted my 3D model to be oriented, so that I could have a great visual to refer to (sketching is always a great aide for me). Below is the layout of the human body that I sketched and followed.

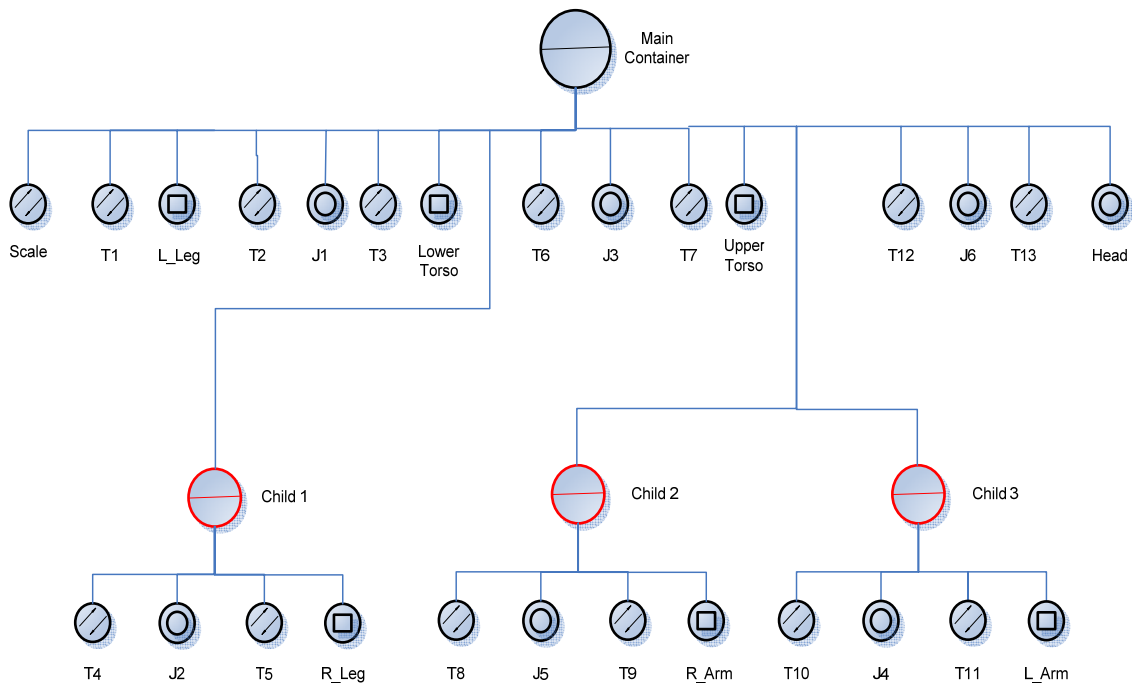


Figure 4: Structural system of body in program terms

In this diagram I specified where and how I wanted each main body part to be oriented. I had to specify the actual radius, length, and width of most of these body parts. My goal was to have a 3D model that closely represented the same dimensions of that of a human body, having body parts being similar in relativity. While sketching my first body

diagram (because it took multiple attempts before I was satisfied with my end result) I realized that something was missing. The body didn't seem to be as mobile and free flowing as an actual human body and that was because the body parts were so rigid and they didn't seem to smoothly integrate with each other. So I decided to add small joints to the body diagram, which would add freer movements and also allowed the attached body parts to move relative to this joint versus relative to an entire body part. The addition of these six joints (between the legs and the lower torso, between the upper and lower torso, between the arms and the upper torso, and between the head and the upper torso) extremely increased the mobility and flexibility of my model. These joints were exactly what my 3D model was missing. With this addition the body moved more freely and with more grace, similar to that of a real human body. So, now that I was pleased with the current layout of my body I could move on to the next stage of this process which was the actual software structure of the body's layout. This was a diagram that would come in handy when I actually building the program for my model. It allowed me to visually refer to how the body was connected in terms of the Coin3D software. Having this system layout was very beneficial for me so that whenever I was in doubt of how the body components were connected I could simply refer to this diagram. Below is a the actual body structural system viewed by the program.

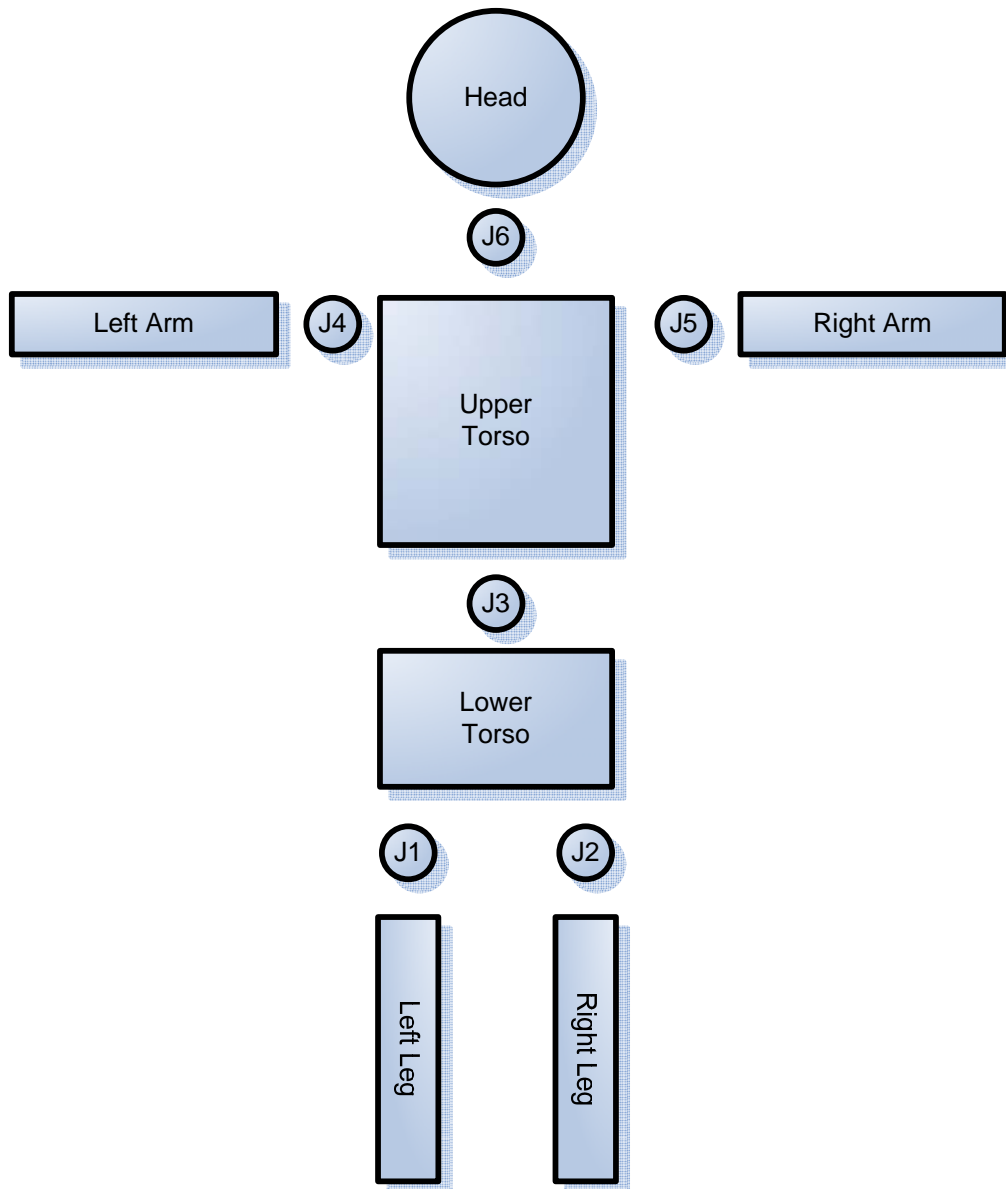


Figure 5: Body Layout

After having this diagram completed, it enabled the task of writing the code for this program a little more doable, even though the code itself was rather lengthy. Another thing I wanted to mention was that the Coin3D program worked only with the kinematics

of the system and not the dynamics. So I could build the body any way I wanted it to appear and I can control how the body moved (i.e. which axis it rotated on, where I wanted the body parts to be located, etc.) and for simplicity I had the body parts rotate about the z-axis only. I could have just as easily had it move about the x or y-axis or any combination of axes. You can refer to Appendix A to see the entire program written in Coin3D for the construction of the 3D human model.

3.4 Generating joint angles in Matlab

The last programming step I had to complete in order for the entire system that I develop to function properly was the code for generating the joint angles, in Matlab, which ultimately caused the human body to move (refer to Appendix B). Depending on how I wanted the specific body parts to move dictated what trigonometry functions I used. For most of the body parts I used sinusoidal movements because the body mainly moves periodically, frequently repeating movements over and over. After specifying the signals that I wanted to send to the joint arrays in Matlab I had an output text file (refer to Appendix C) created, opened, and had the joint angles values stored accordingly. Below are the six signals I generated for the movement of the 3D model.

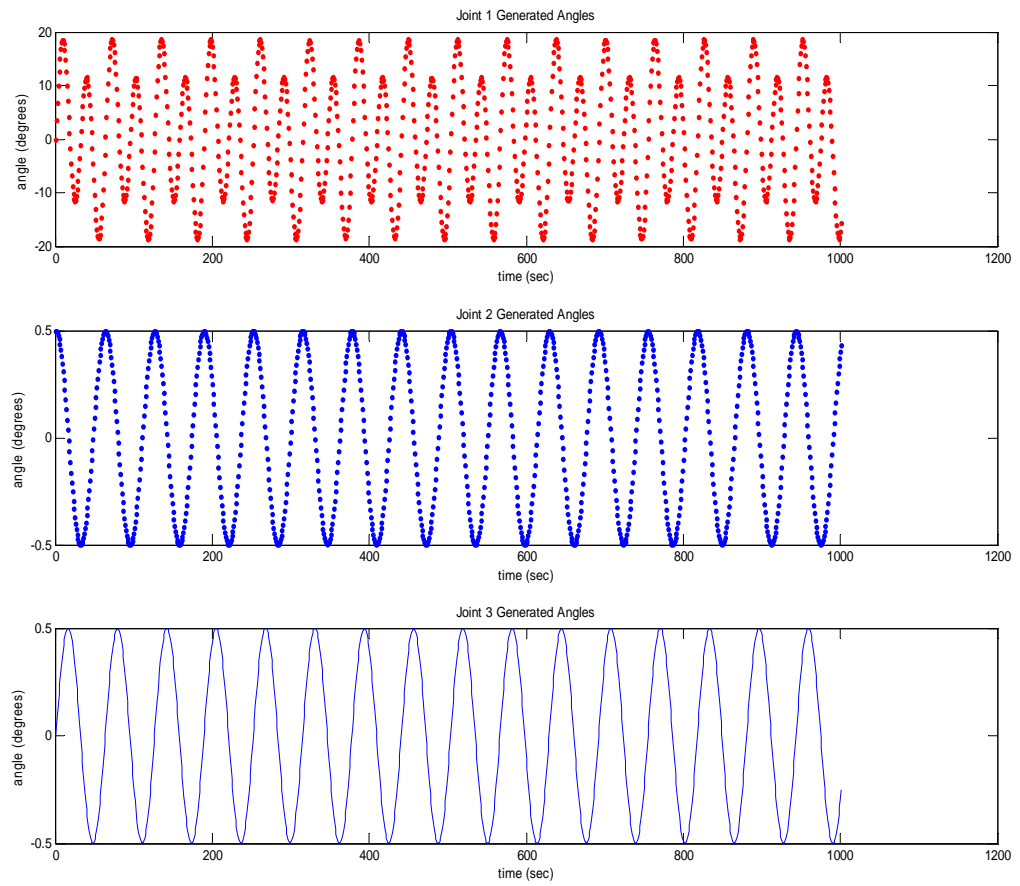


Figure 6: Generated signals for joints 1, 2, & 3

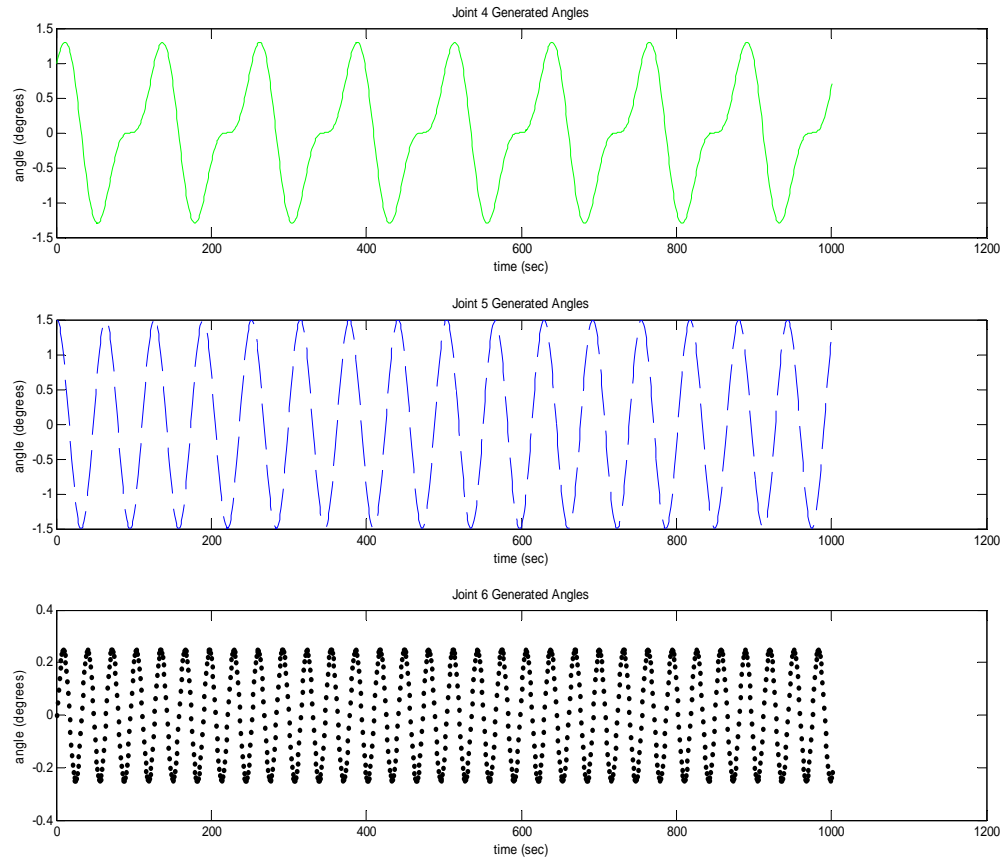


Figure 7: Generated signals for joints 4, 5, & 6

As state before most of these generated signals are very sinusoidal in nature because I wanted to emulate the periodic movement of humans, which result from hearing music. This text file was then copied and inserted in the same directory as the program. Inside of the code for the 3D model I had a command that opened the saved text file and inserted the values into the necessary joint angle transform nodes.

CHAPTER 4

EXPERIMENTAL RESULTS

4.1 Results

After successfully completing each component of the system independently, I began to integrate each component of the system until the entire system was integrated. So, I connected the Matlab program, that generates the joint angles, to the program I wrote in the Coin3D software, and next I inserted the counter, that counted up and selected the actual joint angles within the joint arrays, into to body program. Even though each component was completed and working independently I still had to change a few parameters within the setup of the code so that all components of the system would mesh together. Below is an actual screen shot of the 3D human model I constructed using the Coin3D software.

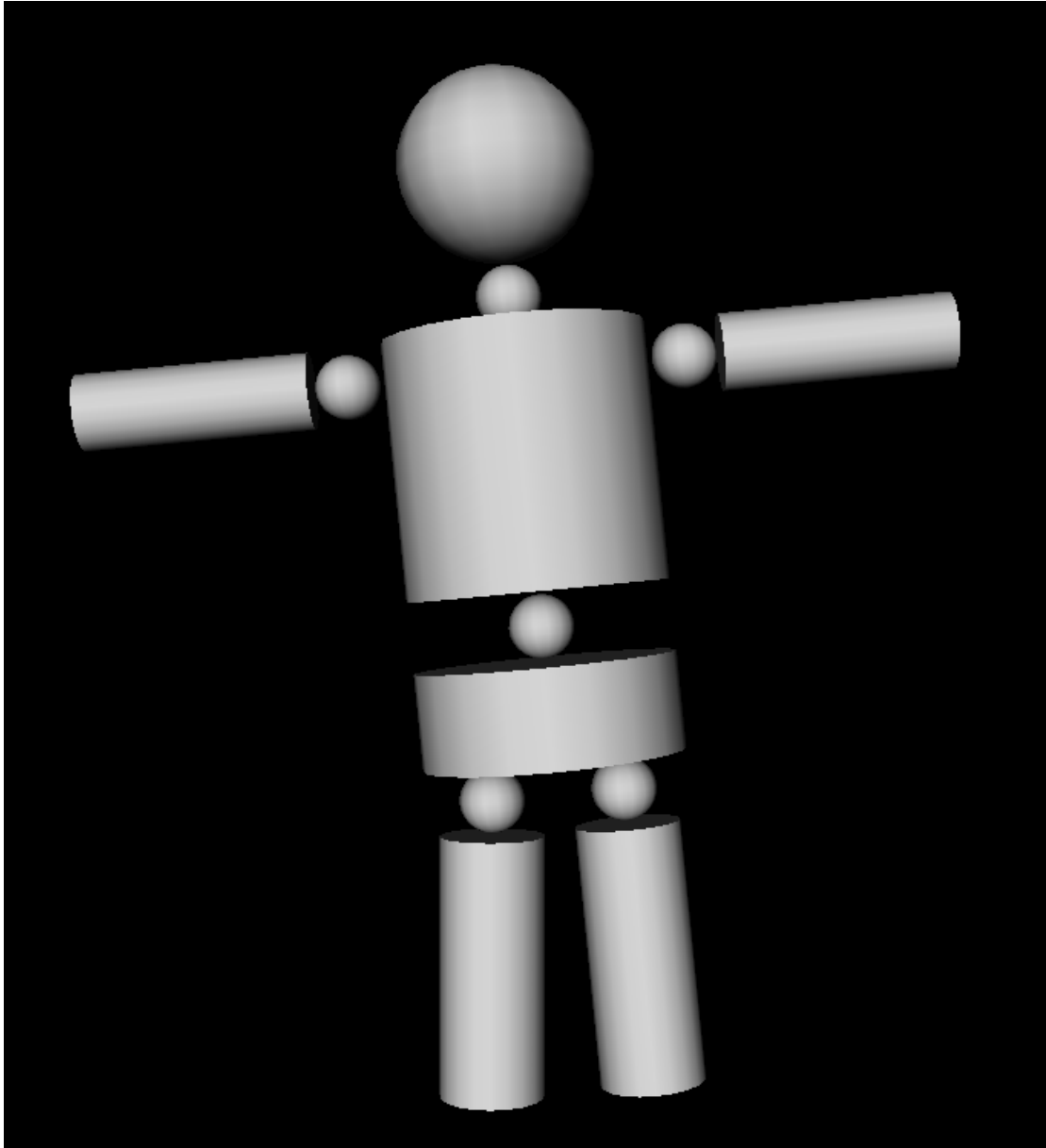


Figure 8: Actual 3D body model

As you can see the actual 3D model closely emanates the body layout that I sketched prior to writing the program for this 3D model. Its ability to closely emulate

the exact shapes and structures that you want is exactly why I selected to use this Coin3D software. The only disadvantage of this program is that I was unable to figure out how to control the dynamics of the model I created but this can be discussed in the recommendations section. I have also recorded the 3D model dancing after running the program and this short movie will be available to view on Professor Hooshang Hemami's website. Here is a link to his website: <http://www.ece.osu.edu/%7Ehemami/>

4.2 Complications

Strange enough with a project like this, which consisted of a lot of programming, I didn't run into many complications. I encountered a little difficulty integrating the entire system together but once I noticed what changes I needed to make everything worked smoothly. The biggest challenge was being able to meet my stretch goal of actually sending in a real music string into Matlab and then dissecting and filtering the music, ending up with a signal that I can then use to generate the necessary joint angles. I also ran into a slight problem with increasing the frame time of the movements. At first I was only using 100 inputs, causing the 3D model to be jumpy when the frame recycled. After some time I figured out a way to increase the number of inputs to 1000 but this increased the frequency of the movements of the model, causing the model to have irregular body movements. I had to then figure out a way to adjust the frequency so that the frequency of the movements would decrease. It took me a while to figure this problem out but after some time I was able to adjust the frequency so that the body movements would return

back to their normal frequency of movement. So overall I can say that I ran into an average amount of problems but there weren't any serious complications that held up my work for an extended amount of time.

CHAPTER 5

CONCLUSION/ RECOMMENDATIONS

4.1 Summary of completed work

As stated above I was successfully able to complete each component of the system that I had designed and I was then able to integrate each component that I had built independently. I believe I experienced so much success with this project because I took the necessary time to complete the preliminary steps that consisted of constructing two Gantt charts, a system outline, the structural outline viewed by the software, and the 3D model design layout. Completing these preliminary steps enabled me to work through each step with a very clear visual aide. I'm a visual person and these charts and diagrams were exactly what I needed, I could not have imagined successfully completing this project without those diagrams and figures. After integrating the entire system I was astound that the program actually worked and I was able to view my 3D model dance. I believe my efforts for creating this 3D model was definitely worth it. Now that my project is complete I feel I was able to meet the objectives that I set forth at the beginning of this project. And I hope my project benefits those who are interested in learning more about how the brain sends interpreted music signals to the rest of the body for

synchronous movements. My program is basically the shell design of a simulation on how signals are passed throughout the body. If an interested individual wanted to run a series of test on how different signals affected the body, my model would be a great tool to use.

4.2 Future work

I think it would be great to see someone take the program that I developed and made it more sophisticated by sending in a real music file into Matlab to be filtered and the output signal be used to calculate the angles which control the actual movement of the body. This was one of my stretch goals but with the limited time that I had available for this project I was unable to accomplish this goal. I would love to see if someone else could add this component to the rest of the system. Also if someone was interested, it would be great to see dynamics (add resistance to movements based on position, add weight to body parts, etc.) added to this program. To see the body respond more human like would also make this program more beneficial and useful for testing purposes.

Bibliography

Hodges, Donald A. *Music and the Brain: Implications on Music and the Brain Research*. September 2000. Music Editors Journal. February 2007.
<http://www.jstor.org/view/00274321/sp060452/06x2779m/0?frame=noframe&userID=a46ba50c@ohio-state.edu/01cc99332100501b81747&dpi=3&config=jstor>

Appendix A: Coin3D program

```
//actual torso
#include <Inventor/Win/SoWin.h>
#include <Inventor/Win/viewers/SoWinExaminerViewer.h>
#include <Inventor/nodes/SoSeparator.h>
#include <Inventor/nodes/SoSphere.h>
#include <Inventor/nodes/SoCylinder.h>
#include <Inventor/nodes/SoCube.h>
#include <Inventor/nodes/SoTransform.h>

#include <Inventor/SoInput.h>
#include <Inventor/fields/SoMFFloat.h>
#include <Inventor/engines/SoTimeCounter.h>
#include <Inventor/engines/SoSelectOne.h>
#include <Inventor/engines/SoComposeRotation.h>

#include <iostream>
using namespace std;

int main(int argc, char ** argv) {
    //*****don't touch
    //create the window - all programs will have this bit of code
    HWND window = SoWin::init(argv[0]);
    if (window==NULL) exit(1); //note that window is a variable name
    SoWinExaminerViewer *viewer = new SoWinExaminerViewer(window);
    //note that viewer is a variable name
    //*****end of don't touch
    //body part parameters
    float joint_radius=1.0f;
    float joint_space=0.1f;
    float leg_height=8.0f;
    float leg_radius=1.6f;
    float lower_torso_height=3.0f;
    float lower_torso_radius=4.0f;
    float upper_torso_height=8.0f;
    float upper_torso_radius=4.0f;
    float rarm_height=7.0f;
    float rarm_radius=1.2f;
    float larm_height=7.0f;
    float larm_radius=1.2f;
    float head_radius=3.0f;
    float sc=1.0f;

    //body parts, joints and transform declarations
    SoSeparator *container = new SoSeparator;
    SoTransform *scale = new SoTransform;
    SoTransform *t1 = new SoTransform;
    SoTransform *t2 = new SoTransform;
    SoTransform *t3 = new SoTransform;
```

```

SoTransform *t4 = new SoTransform;
SoTransform *t5 = new SoTransform;
SoTransform *t6= new SoTransform;
SoTransform *t7= new SoTransform;
SoTransform *t8= new SoTransform;
SoTransform *t9= new SoTransform;
SoTransform *t10= new SoTransform;
SoTransform *t11= new SoTransform;
SoTransform *t12= new SoTransform;
SoTransform *t13= new SoTransform;
SoCylinder *lleg = new SoCylinder;
SoCylinder *rleg = new SoCylinder;
SoCylinder *rarm = new SoCylinder;
SoCylinder *larm = new SoCylinder;
SoSphere *j1=new SoSphere;
SoSphere *j2=new SoSphere;
SoSphere *j3=new SoSphere;
SoSphere *j4=new SoSphere;
SoSphere *j5=new SoSphere;
SoSphere *j6=new SoSphere;
SoCylinder *lower_torso=new SoCylinder;
SoCylinder *upper_torso=new SoCylinder;
SoSeparator *rleg_separator=new SoSeparator;
SoSeparator *rarm_separator=new SoSeparator;
SoSeparator *larm_separator=new SoSeparator;
SoSphere *head=new SoSphere;

//body part scaling
lleg->height.setValue(leg_height);
lleg->radius.setValue(leg_radius);
rleg->height.setValue(leg_height);
rleg->radius.setValue(leg_radius);
lower_torso->height.setValue(lower_torso_height);
lower_torso->radius.setValue(lower_torso_radius);
upper_torso->height.setValue(upper_torso_height);
upper_torso->radius.setValue(upper_torso_radius);
rarm->height.setValue(rarm_height);
rarm->radius.setValue(rarm_radius);
larm->height.setValue(larm_height);
larm->radius.setValue(larm_radius);
j1->radius.setValue(joint_radius);
j2->radius.setValue(joint_radius);
j3->radius.setValue(joint_radius);
j4->radius.setValue(joint_radius);
j5->radius.setValue(joint_radius);
j6->radius.setValue(joint_radius);
head->radius.setValue(head_radius);

//transforms
scale->scaleFactor.setValue(sc,sc,sc);

```

```

        t1->translation.setValue(0.0f,0.5f*leg_height,0.0f);
        t2-
>translation.setValue(0.0f,0.5f*leg_height+joint_space+joint_radius,0.0
f);
        t3->translation.setValue(lower_torso_radius-
2.0f*joint_radius, joint_radius+joint_space+0.5f*lower_torso_height,0.0f
);
        t4->translation.setValue(lower_torso_radius-2.0f*joint_radius,-
(joint_radius+joint_space+0.5f*lower_torso_height),0.0f);
        t5->translation.setValue(0.0f,-
(0.5f*leg_height+joint_space+joint_radius),0.0f);
        t6->translation.setValue(0.0f,
0.5f*lower_torso_height+joint_space+joint_radius, 0.0f);
        t7->translation.setValue(0.0f,
joint_radius+joint_space+0.5f*upper_torso_height, 0.0f);
        t8-
>translation.setValue(upper_torso_radius+joint_radius+joint_space,
0.35f*upper_torso_height, 0.0f);
        t9->translation.setValue(-
(upper_torso_radius+joint_radius+joint_space),
0.35f*upper_torso_height, 0.0f);
        t10-
>translation.setValue(joint_radius+joint_space+0.5f*rarm_height,0.0f,0.
0f);
        t11->translation.setValue(-
(joint_radius+joint_space+0.5f*rarm_height),0.0f,0.0f);
        t12-
>translation.setValue(0.0f,0.5f*upper_torso_height+joint_space+joint_ra
dius,0.0f);
        t13-
>translation.setValue(0.0f, joint_radius+joint_space+head_radius,0.0f);

//rotations
t10->rotation.setValue(SbVec3f(0,0,1),3.14f/2);
t11->rotation.setValue(SbVec3f(0,0,1),3.14f/2);

//construction
container->addChild(scale);
container->addChild(t1);
container->addChild(lleg);
container->addChild(t2);
container->addChild(j1);
container->addChild(t3);
container->addChild(lower_torso);
container->addChild(rleg_separator);
container->addChild(t6);
container->addChild(j3);
container->addChild(t7);
container->addChild(upper_torso);
container->addChild(rarm_separator);

```

```

container->addChild(larm_separator);
container->addChild(t12);
container->addChild(j6);
container->addChild(t13);
container->addChild(head);

//extra separators
rleg_separator->addChild(t4);
rleg_separator->addChild(j2);
rleg_separator->addChild(t5);
rleg_separator->addChild(rleg);

rarm_separator->addChild(t8);
rarm_separator->addChild(j5);
rarm_separator->addChild(t10);
rarm_separator->addChild(rarm);

larm_separator->addChild(t9);
larm_separator->addChild(j4);
larm_separator->addChild(t11);
larm_separator->addChild(larm);

//read a file
SoInput in;
if(in.openFile("joint_angles_test.txt")) {
    cout << "open sucessful" << endl;
}else {
    cout << "could not read file..." << endl;
}

//make storage space for angles
SoMFFloat j1_vals;
SoMFFloat j2_vals;
SoMFFloat j3_vals;
SoMFFloat j4_vals;
SoMFFloat j5_vals;
SoMFFloat j6_vals;

//number of joint angle entries
int num_entries=100;

/*read the file into storage variables*/
float joint_val=0.0f;
for(int i=0;i<num_entries;i++){
    in.read(joint_val);
    j1_vals.set1Value(i,joint_val);

    in.read(joint_val);
    j2_vals.set1Value(i,joint_val);

```

```

        in.read(joint_val);
        j3_vals.set1Value(i,joint_val);

        in.read(joint_val);
        j4_vals.set1Value(i,joint_val);

        in.read(joint_val);
        j5_vals.set1Value(i,joint_val);

        in.read(joint_val);
        j6_vals.set1Value(i,joint_val);
    }
    cout << "read successful" << endl;
    /*storage variables j1,j2,...,j6 should have joint angles
    in them*/

    //uncomment to print out of joint angles
    /*
    for(int i=0;i<num_entries;i++){
        cout << j1_vals[i] << " " << j2_vals[i] << " " \
            << j3_vals[i] << " " << j4_vals[i] << " " \
            << j5_vals[i] << " " << j6_vals[i] << endl;
    }
    */

    //make a timing source and set the inputs
    SoTimeCounter *myCounter = new SoTimeCounter;
    myCounter->min.setValue(0);
    myCounter->max.setValue(num_entries-1);
    myCounter->frequency.setValue(0.2); //timestep

    //make the selection engines
    SoSelectOne *j1_select=new
SoSelectOne(SoMFFloat::getClassTypeId());
    SoSelectOne *j2_select=new
SoSelectOne(SoMFFloat::getClassTypeId());
    SoSelectOne *j3_select=new
SoSelectOne(SoMFFloat::getClassTypeId());
    SoSelectOne *j4_select=new
SoSelectOne(SoMFFloat::getClassTypeId());
    SoSelectOne *j5_select=new
SoSelectOne(SoMFFloat::getClassTypeId());
    SoSelectOne *j6_select=new
SoSelectOne(SoMFFloat::getClassTypeId());

    //associate each selection engine with a joints value array
    j1_select->input=&j1_vals;
    j2_select->input=&j2_vals;
    j3_select->input=&j3_vals;
    j4_select->input=&j4_vals;

```



```

j5_select->input=&j5_vals;
j6_select->input=&j6_vals;

//connect index input of selection engines to output of timer
j1_select->index.connectFrom(&myCounter->output);
j2_select->index.connectFrom(&myCounter->output);
j3_select->index.connectFrom(&myCounter->output);
j4_select->index.connectFrom(&myCounter->output);
j5_select->index.connectFrom(&myCounter->output);
j6_select->index.connectFrom(&myCounter->output);

//make the compose rotation engines
SoComposeRotation *j1r_engine = new SoComposeRotation;
SoComposeRotation *j2r_engine = new SoComposeRotation;
SoComposeRotation *j3r_engine = new SoComposeRotation;
SoComposeRotation *j4r_engine = new SoComposeRotation;
SoComposeRotation *j5r_engine = new SoComposeRotation;
SoComposeRotation *j6r_engine = new SoComposeRotation;

//set the rotation engine axes
j1r_engine->axis.setValue(0,0,1);
j2r_engine->axis.setValue(0,0,1);
j3r_engine->axis.setValue(0,0,1);
j4r_engine->axis.setValue(0,0,1);
j5r_engine->axis.setValue(0,0,1);
j6r_engine->axis.setValue(0,0,1);

//set the rotation engine angles
j1r_engine->angle.connectFrom(j1_select->output);
j2r_engine->angle.connectFrom(j2_select->output);
j3r_engine->angle.connectFrom(j3_select->output);
j4r_engine->angle.connectFrom(j4_select->output);
j5r_engine->angle.connectFrom(j5_select->output);
j6r_engine->angle.connectFrom(j6_select->output);

//connect selection engine outputs to joint transform angles
t2->rotation.connectFrom(&j1r_engine->rotation);
t4->rotation.connectFrom(&j2r_engine->rotation);
t6->rotation.connectFrom(&j3r_engine->rotation);
t8->rotation.connectFrom(&j5r_engine->rotation);
t9->rotation.connectFrom(&j4r_engine->rotation);
t12->rotation.connectFrom(&j6r_engine->rotation);

cout << "my program makes it here" << endl;

//*****don't touch
//tell the viewer what scenegraph to show
viewer->setSceneGraph(container);
viewer->show();

```

```
code    //clean up what you have made - all program will have this bit of
        SoWin::show(window);
        SoWin::mainLoop();
        delete viewer;

        return 0;
        //*****end of don't touch
    }
```

Appendix B: Joint Angle Generation Program in Matlab

```
clear all;
close all;

tstep=0.1; %sec
t=0:tstep:100;

%angles in degrees
j1=5*sin(t)+15*sin(2*t);
j2=0.5*cos(t);
j3=0.5*sin(t);
j4=cos(0.5*t)+ 0.5*sin(t);
j5=1.5*cos(t);
j6=0.25*sin(2*t);

%plot the signals

subplot(3,1,1)
plot(j1, 'r')
title('Joint 1 Generated Angles')
xlabel('time (sec)')
ylabel('angle (degrees)')
subplot(3,1,2)
plot(j2, '.')
title('Joint 2 Generated Angles')
xlabel('time (sec)')
ylabel('angle (degrees)')
subplot(3,1,3)
plot(j3, '-')
title('Joint 3 Generated Angles')
xlabel('time (sec)')
ylabel('angle (degrees)')

Subplot(3,1,1)
plot(j4, '-g')
title('Joint 4 Generated Angles')
xlabel('time (sec)')
ylabel('angle (degrees)')
subplot(3,1,2)
plot(j5, '--b')
title('Joint 5 Generated Angles')
xlabel('time (sec)')
ylabel('angle (degrees)')
subplot(3,1,3)
plot(j6, '.k')
title('Joint 6 Generated Angles')
xlabel('time (sec)')
ylabel('angle (degrees)')
```

```
%convert to radians
j1=j1*pi/180;

%fid=fopen('joint_angles_test.txt','w');
fid=fopen('Z:\Distinction
Project\Coin3DTest2005\debug\joint_angles_test.txt','w');
for(i=1:length(t))
    fprintf(fid,'%d %d %d %d %d
%d\n',j1(i),j2(i),j3(i),j4(i),j5(i),j6(i));
end
fclose(fid);
```

Appendix C: Sample text file of generated angles

```
0 0 1 1.224647e-016 1 0
9.983342e-002 1.986693e-001 9.950042e-001 -9.983342e-002 9.950042e-001
0
1.986693e-001 3.894183e-001 9.800666e-001 -1.986693e-001 9.800666e-001
0
2.955202e-001 5.646425e-001 9.553365e-001 -2.955202e-001 9.553365e-001
0
3.894183e-001 7.173561e-001 9.210610e-001 -3.894183e-001 9.210610e-001
0
4.794255e-001 8.414710e-001 8.775826e-001 -4.794255e-001 8.775826e-001
0
5.646425e-001 9.320391e-001 8.253356e-001 -5.646425e-001 8.253356e-001
0
6.442177e-001 9.854497e-001 7.648422e-001 -6.442177e-001 7.648422e-001
0
7.173561e-001 9.995736e-001 6.967067e-001 -7.173561e-001 6.967067e-001
0
7.833269e-001 9.738476e-001 6.216100e-001 -7.833269e-001 6.216100e-001
0
8.414710e-001 9.092974e-001 5.403023e-001 -8.414710e-001 5.403023e-001
0
8.912074e-001 8.084964e-001 4.535961e-001 -8.912074e-001 4.535961e-001
0
9.320391e-001 6.754632e-001 3.623578e-001 -9.320391e-001 3.623578e-001
0
9.635582e-001 5.155014e-001 2.674988e-001 -9.635582e-001 2.674988e-001
0
9.854497e-001 3.349882e-001 1.699671e-001 -9.854497e-001 1.699671e-001
0
9.974950e-001 1.411200e-001 7.073720e-002 -9.974950e-001 7.073720e-002
0
9.995736e-001 -5.837414e-002 -2.919952e-002 -9.995736e-001 -2.919952e-
002 0
9.916648e-001 -2.555411e-001 -1.288445e-001 -9.916648e-001 -1.288445e-
001 0
9.738476e-001 -4.425204e-001 -2.272021e-001 -9.738476e-001 -2.272021e-
001 0
9.463001e-001 -6.118579e-001 -3.232896e-001 -9.463001e-001 -3.232896e-
001 0
9.092974e-001 -7.568025e-001 -4.161468e-001 -9.092974e-001 -4.161468e-
001 0
8.632094e-001 -8.715758e-001 -5.048461e-001 -8.632094e-001 -5.048461e-
001 0
8.084964e-001 -9.516021e-001 -5.885011e-001 -8.084964e-001 -5.885011e-
001 0
7.457052e-001 -9.936910e-001 -6.662760e-001 -7.457052e-001 -6.662760e-
001 0
```

6.754632e-001 -9.961646e-001 -7.373937e-001 -6.754632e-001 -7.373937e-001 0
5.984721e-001 -9.589243e-001 -8.011436e-001 -5.984721e-001 -8.011436e-001 0
5.155014e-001 -8.834547e-001 -8.568888e-001 -5.155014e-001 -8.568888e-001 0
4.273799e-001 -7.727645e-001 -9.040721e-001 -4.273799e-001 -9.040721e-001 0
3.349882e-001 -6.312666e-001 -9.422223e-001 -3.349882e-001 -9.422223e-001 0
2.392493e-001 -4.646022e-001 -9.709582e-001 -2.392493e-001 -9.709582e-001 0
1.411200e-001 -2.794155e-001 -9.899925e-001 -1.411200e-001 -9.899925e-001 0
4.158066e-002 -8.308940e-002 -9.991352e-001 -4.158066e-002 -9.991352e-001 0
-5.837414e-002 1.165492e-001 -9.982948e-001 5.837414e-002 -9.982948e-001 0
-1.577457e-001 3.115414e-001 -9.874798e-001 1.577457e-001 -9.874798e-001 0
-2.555411e-001 4.941134e-001 -9.667982e-001 2.555411e-001 -9.667982e-001 0
-3.507832e-001 6.569866e-001 -9.364567e-001 3.507832e-001 -9.364567e-001 0
-4.425204e-001 7.936679e-001 -8.967584e-001 4.425204e-001 -8.967584e-001 0
-5.298361e-001 8.987081e-001 -8.481000e-001 5.298361e-001 -8.481000e-001 0
-6.118579e-001 9.679197e-001 -7.909677e-001 6.118579e-001 -7.909677e-001 0
-6.877662e-001 9.985433e-001 -7.259323e-001 6.877662e-001 -7.259323e-001 0
-7.568025e-001 9.893582e-001 -6.536436e-001 7.568025e-001 -6.536436e-001 0
-8.182771e-001 9.407306e-001 -5.748239e-001 8.182771e-001 -5.748239e-001 0
-8.715758e-001 8.545989e-001 -4.902608e-001 8.715758e-001 -4.902608e-001 0
-9.161659e-001 7.343971e-001 -4.007992e-001 9.161659e-001 -4.007992e-001 0
-9.516021e-001 5.849172e-001 -3.073329e-001 9.516021e-001 -3.073329e-001 0
-9.775301e-001 4.121185e-001 -2.107958e-001 9.775301e-001 -2.107958e-001 0
-9.936910e-001 2.228899e-001 -1.121525e-001 9.936910e-001 -1.121525e-001 0
-9.999233e-001 2.477543e-002 -1.238866e-002 9.999233e-001 -1.238866e-002 0
-9.961646e-001 -1.743268e-001 8.749898e-002 9.961646e-001 8.749898e-002 0

-9.824526e-001 -3.664791e-001 1.865124e-001 9.824526e-001 1.865124e-001
0
-9.589243e-001 -5.440211e-001 2.836622e-001 9.589243e-001 2.836622e-001
0
-9.258147e-001 -6.998747e-001 3.779777e-001 9.258147e-001 3.779777e-001
0
-8.834547e-001 -8.278265e-001 4.685167e-001 8.834547e-001 4.685167e-001
0
-8.322674e-001 -9.227754e-001 5.543743e-001 8.322674e-001 5.543743e-001
0
-7.727645e-001 -9.809362e-001 6.346929e-001 7.727645e-001 6.346929e-001
0
-7.055403e-001 -9.999902e-001 7.086698e-001 7.055403e-001 7.086698e-001
0
-6.312666e-001 -9.791777e-001 7.755659e-001 6.312666e-001 7.755659e-001
0
-5.506855e-001 -9.193285e-001 8.347128e-001 5.506855e-001 8.347128e-001
0
-4.646022e-001 -8.228286e-001 8.855195e-001 4.646022e-001 8.855195e-001
0
-3.738767e-001 -6.935251e-001 9.274784e-001 3.738767e-001 9.274784e-001
0
-2.794155e-001 -5.365729e-001 9.601703e-001 2.794155e-001 9.601703e-001
0
-1.821625e-001 -3.582293e-001 9.832684e-001 1.821625e-001 9.832684e-001
0
-8.308940e-002 -1.656042e-001 9.965421e-001 8.308940e-002 9.965421e-001
0
1.681390e-002 3.362305e-002 9.998586e-001 -1.681390e-002 9.998586e-001
0
1.165492e-001 2.315098e-001 9.931849e-001 -1.165492e-001 9.931849e-001
0
2.151200e-001 4.201670e-001 9.765876e-001 -2.151200e-001 9.765876e-001
0
3.115414e-001 5.920735e-001 9.502326e-001 -3.115414e-001 9.502326e-001
0
4.048499e-001 7.403759e-001 9.143831e-001 -4.048499e-001 9.143831e-001
0
4.941134e-001 8.591618e-001 8.693975e-001 -4.941134e-001 8.693975e-001
0
5.784398e-001 9.436957e-001 8.157251e-001 -5.784398e-001 8.157251e-001
0
6.569866e-001 9.906074e-001 7.539023e-001 -6.569866e-001 7.539023e-001
0
7.289690e-001 9.980267e-001 6.845467e-001 -7.289690e-001 6.845467e-001
0
7.936679e-001 9.656578e-001 6.083513e-001 -7.936679e-001 6.083513e-001
0
8.504366e-001 8.947912e-001 5.260775e-001 -8.504366e-001 5.260775e-001
0

8.987081e-001 7.882521e-001 4.385473e-001 -8.987081e-001 4.385473e-001
0
9.380000e-001 6.502878e-001 3.466353e-001 -9.380000e-001 3.466353e-001
0
9.679197e-001 4.863987e-001 2.512598e-001 -9.679197e-001 2.512598e-001
0
9.881682e-001 3.031184e-001 1.533739e-001 -9.881682e-001 1.533739e-001
0
9.985433e-001 1.077537e-001 5.395542e-002 -9.985433e-001 5.395542e-002
0
9.989413e-001 -9.190685e-002 -4.600213e-002 -9.989413e-001 -4.600213e-
002 0
9.893582e-001 -2.879033e-001 -1.455000e-001 -9.893582e-001 -1.455000e-
001 0
9.698898e-001 -4.724220e-001 -2.435442e-001 -9.698898e-001 -2.435442e-
001 0
9.407306e-001 -6.381067e-001 -3.391549e-001 -9.407306e-001 -3.391549e-
001 0
9.021718e-001 -7.783521e-001 -4.313768e-001 -9.021718e-001 -4.313768e-
001 0
8.545989e-001 -8.875670e-001 -5.192887e-001 -8.545989e-001 -5.192887e-
001 0
7.984871e-001 -9.613975e-001 -6.020119e-001 -7.984871e-001 -6.020119e-
001 0
7.343971e-001 -9.969001e-001 -6.787200e-001 -7.343971e-001 -6.787200e-
001 0
6.629692e-001 -9.926594e-001 -7.486466e-001 -6.629692e-001 -7.486466e-
001 0
5.849172e-001 -9.488445e-001 -8.110930e-001 -5.849172e-001 -8.110930e-
001 0
5.010209e-001 -8.672022e-001 -8.654352e-001 -5.010209e-001 -8.654352e-
001 0
4.121185e-001 -7.509872e-001 -9.111303e-001 -4.121185e-001 -9.111303e-
001 0
3.190984e-001 -6.048328e-001 -9.477216e-001 -3.190984e-001 -9.477216e-
001 0
2.228899e-001 -4.345656e-001 -9.748436e-001 -2.228899e-001 -9.748436e-
001 0
1.244544e-001 -2.469737e-001 -9.922253e-001 -1.244544e-001 -9.922253e-
001 0
2.477543e-002 -4.953564e-002 -9.996930e-001 -2.477543e-002 -9.996930e-
001 0
-7.515112e-002 1.498772e-001 -9.971722e-001 7.515112e-002 -9.971722e-
001 0
-1.743268e-001 3.433149e-001 -9.846879e-001 1.743268e-001 -9.846879e-
001 0
-2.717606e-001 5.230658e-001 -9.623649e-001 2.717606e-001 -9.623649e-
001 0
-3.664791e-001 6.819636e-001 -9.304263e-001 3.664791e-001 -9.304263e-
001 0

-4.575359e-001 8.136737e-001 -8.891912e-001 4.575359e-001 -8.891912e-
001 0
-5.440211e-001 9.129453e-001 -8.390715e-001 5.440211e-001 -8.390715e-
001 0